Introduction to Assimilation of Atmospheric Composition Observations with WRF-Chem

Arthur P. Mizzi

(mizzi@ucar.edu)

303-903-5544

WRF-Chem Tutorial
January 29–31, 2018

Overview

- Introduction to “data assimilation (DA)” and ensemble DA methods.
- Review of chemical data assimilation efforts with WRF-Chem.
- Special challenges with assimilation of observations of atmospheric composition.
- Review of WRF-Chem/DART and the tutorial on localization.

Arthur P. Mizzi
mizzi@ucar.edu

Background: Introduction to Data Assimilation Methods

Arthur P. Mizzi
mizzi@ucar.edu

Data Assimilation Methods

- Ensemble Kalman Filter Methods
  - ✓ EnKF – Ensemble Kalman Filter (Houtekamer and Mitchell, 1998; 2001)
  - ✓ EAKF – Ensemble Adjustment Kalman Filter (Anderson, 2001; 2003)
  - ✓ ETKF – Ensemble Transform Kalman Filter(Bishop et al., 2001)
  - ✓ LETKF – Local Ensemble Transform Kalman Filter (Ott et. al., 2002; 2004)
- Hybrid Methods
  - ✓ Weighted sum (Hamill and Snyder, 2000)
  - ✓ Control variable (Lorenc, 2003)
  - ✓ EVIL – Ensemble Variational Integrated Localizaed hybrid (Aulignes et al., 2016).

Arthur P. Mizzi
mizzi@ucar.edu

Data Assimilation Methods cont.

- Constrained Emissions (Emissions Inversion)
  - ✓Adjoint Method
  - ✓Variational Method
  - ✓State Augmentation Methods

Arthur P. Mizzi
mizzi@ucar.edu

Review of Chemical Data Assimilation Efforts with WRF-Chem

Arthur P. Mizzi
mizzi@ucar.edu

WRF-Chem/Chem DA Efforts

- WRF-Chem/GSI
  - ✓ Liu et al., NCAR/MMM – Assimilate MODIS AOD, $PM_{2.5}$, and PM_{10} surface observations.
  - ✓ Pagowski and Grell, NOAA – Assimilate O_3 and $PM_{2.5}$.
  - ✓ Saide et al., Univ. of Iowa and NCAR/ACOM – Assimilate MODIS AOD with constrained emissions.
- WRF-Chem/EnKF
  - ✓ Pagowski and Grell, NOAA – Assimilate $PM_{2.5}$ surface observations.
- WRF-Chem/DART (EAKF)
  - ✓ Mizzi and Edwards, NCAR/ACOM; Anderson, NCAR/IMAGe; and Arellano, Univ. of Arizona – Assimilate MOPITT and IASI CO profiles and MODIS AOD with constrained emissions.

Arthur P. Mizzi
mizzi@ucar.edu

WRF-Chem/Chem DA Efforts cont.

- WRF-Chem/DART (EAKF) cont.
  - ✓ Mizzi, NCAR/ACOM and Cohen and Liu, Univ. of California at Berkeley – Assimilate OMI total column NO_2 with constrained emissions.
  - ✓ Mizzi, NCAR/ACOM and Chen, Miao, and Liang, York Univ. – Assimilate MOPITT total column CO with constrained emissions and MODIS AOD.
  - ✓ Mizzi, NCAR/ACOM Pfister, and Edwards – Application of WRF-Chem/DART to quasi-real time dual-resolution air quality forecasting/cycling in FRAPPE/DiscoverAQ.
  - ✓ Mizzi, NCAR and Mirzargar, Univ. of Miami – Using data-depth algorithms to identify the most representative ensemble member for quasi-real, time dual-resolution air quality forecasting/assimilation.

WRF-Chem/Chem DA Efforts cont.

- WRF-Chem/DART (EAKF) cont.
  - ✓ Mizzi, NCAR and Brasseur and Bouarar, Max Plank Inst. for Meteorology – Application of WRF-Chem/DART to quasi-real time, dual-resolution air quality forecasting/cycling for eastern China in connection with PANDA.
  - ✓ Mizzi, NCAR and Wang and Ma, Nanjing Univ. –Application of WRF-Chem/DART to quasi-real, time dual-resolution air quality forecasting/cycling for eastern China.
  - ✓ Mizzi, NCAR and Garcia-Reynoso and Sabari, Natl. Autonomous Univ, of Mexico –Application of WRF-Chem/DART to quasi-real, time dual-resolution air quality forecasting/cycling for Mexico City.

WRF-Chem/DART

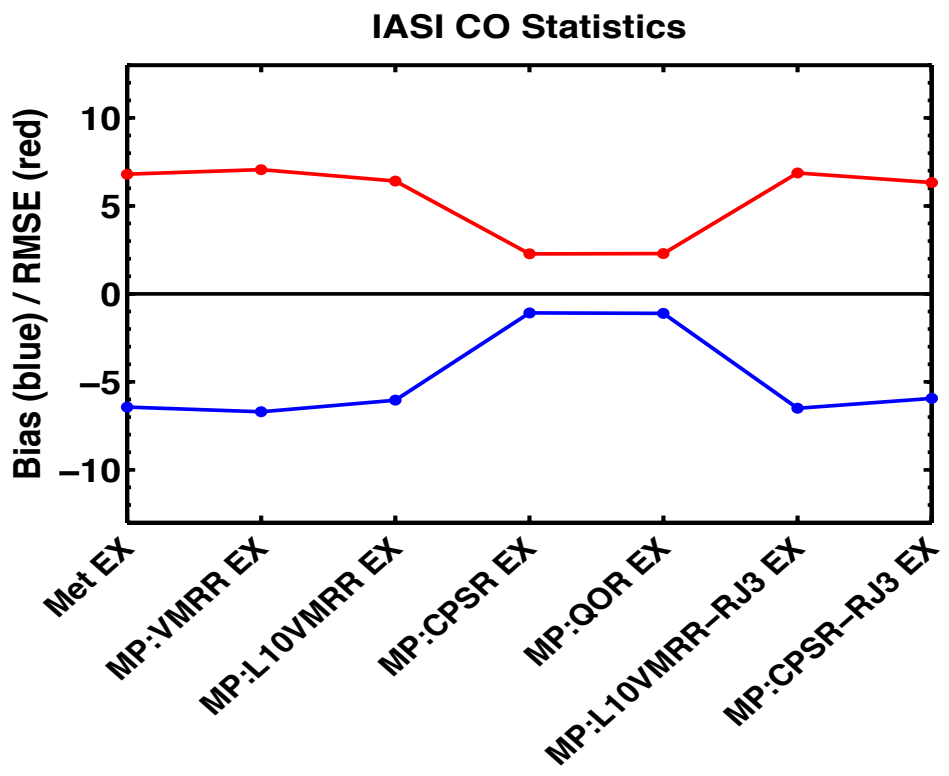
- WRF-Chem – the Weather Research and Forecasting (WRF) model with online chemistry.
- DART – the Data Assimilation Research Testbed modified for assimilation of atmospheric composition observations.
  - MOPITT and IASI partial and total column CO
  - IASI partial and total column O_3 (under testing
  - MODIS AOD and OMI total column NO_2
  - AirNOW and PANDA *in situ* observations
  - Emission constraints – State augmentation method
  - Assimilate as RETRs, QORs, and CPSRs
  - State variable localization (joint or independent assimilation)
  - Quasi-real time and dual-resolution cycling.

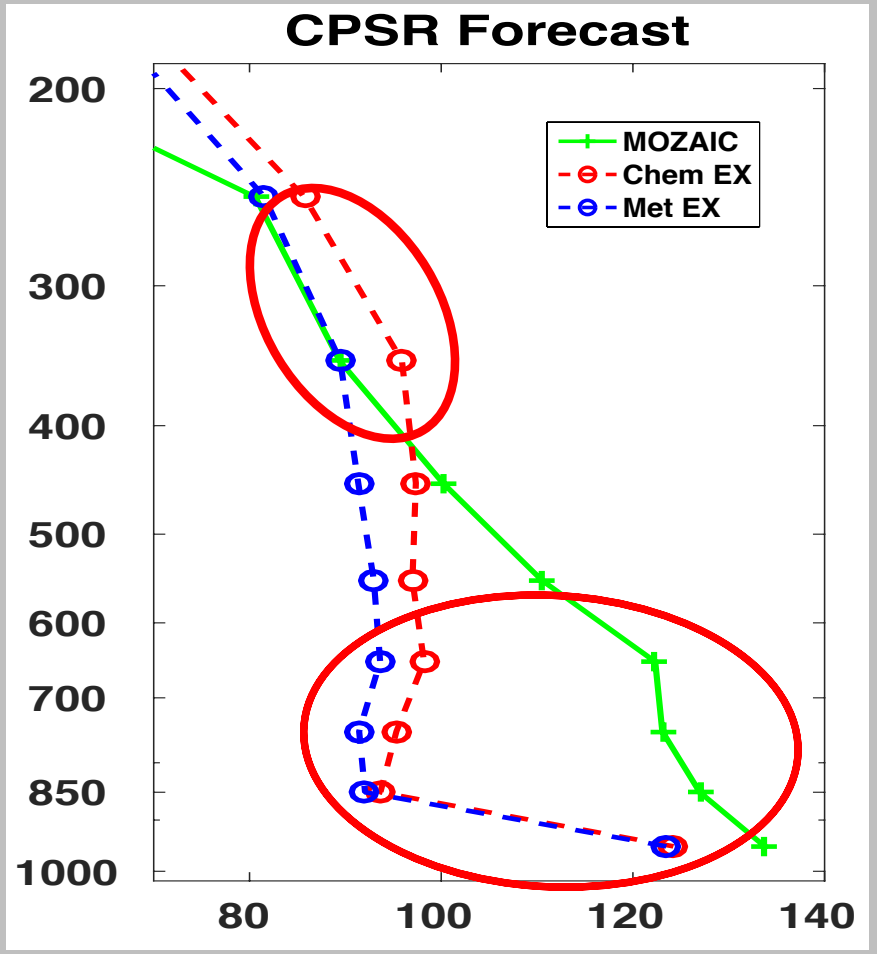
Arthur P. Mizzi
mizzi@ucar.edu

WRF-Chem/DART

- WRF-Chem/DART repository download link: https://svn-dares-dart.cgd.ucar.edu/DART/tags/wrf-chem.r11690
- Sample input/output data:
  - Go to NCAR ftp site at ftp.ucar.edu
  - Go to ../pub/mmm/mizzi
  - Get the PANDA_REAL_TIME_DATA.tar file. This is the input data and directory structure.
  - Get the real_PANDA_RETR_VARLOC.tar file. This is the sample output data and directory structure.
- WRF-Chem/DART Users' Guide is not available so collaboration is recommended/necessary. Contact Dr. Arthur P. Mizzi for information.

2008 Case Study: Forecast Verification

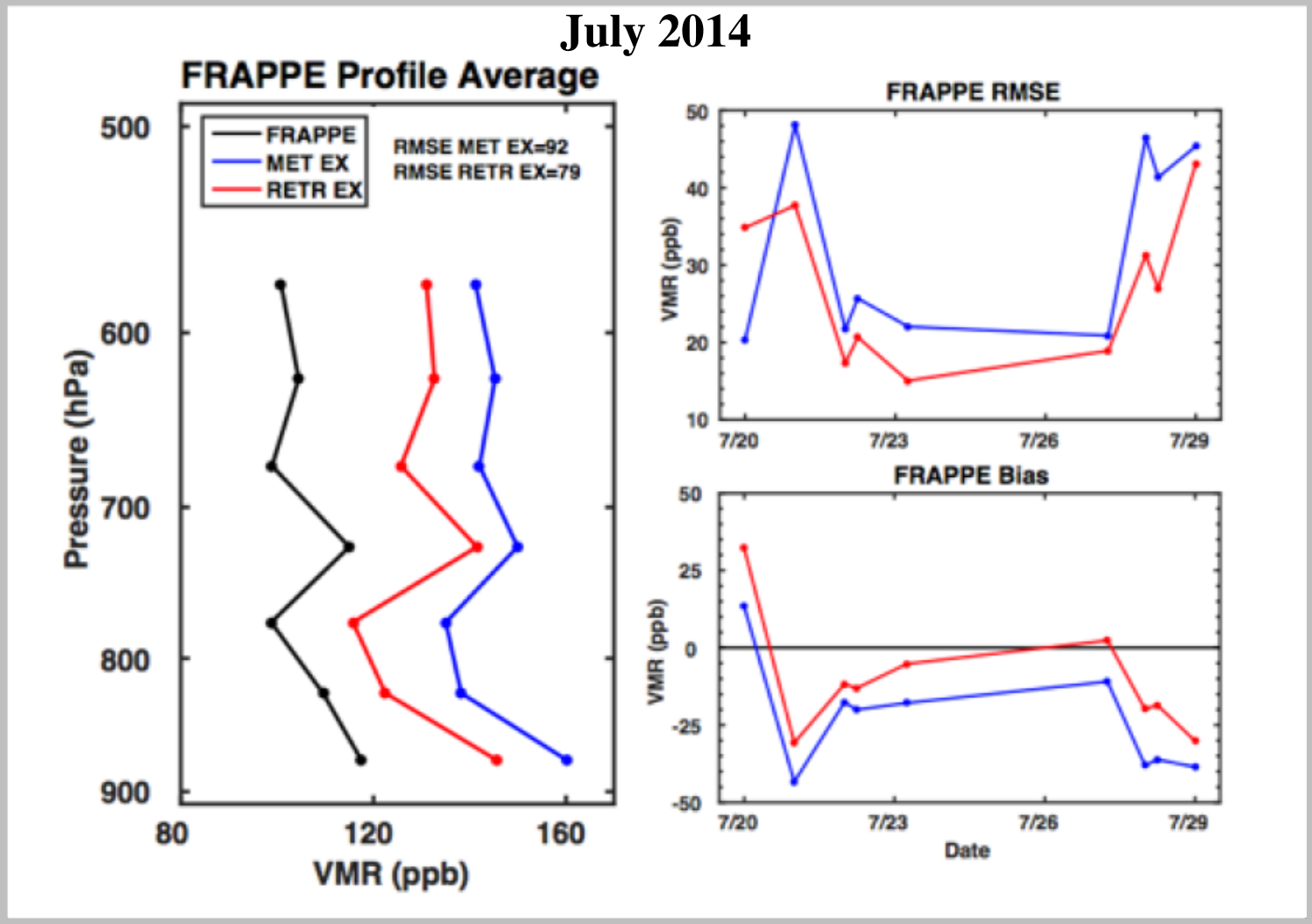




Mizzi et al. (2018)

Arthur P. Mizzi
mizzi@ucar.edu

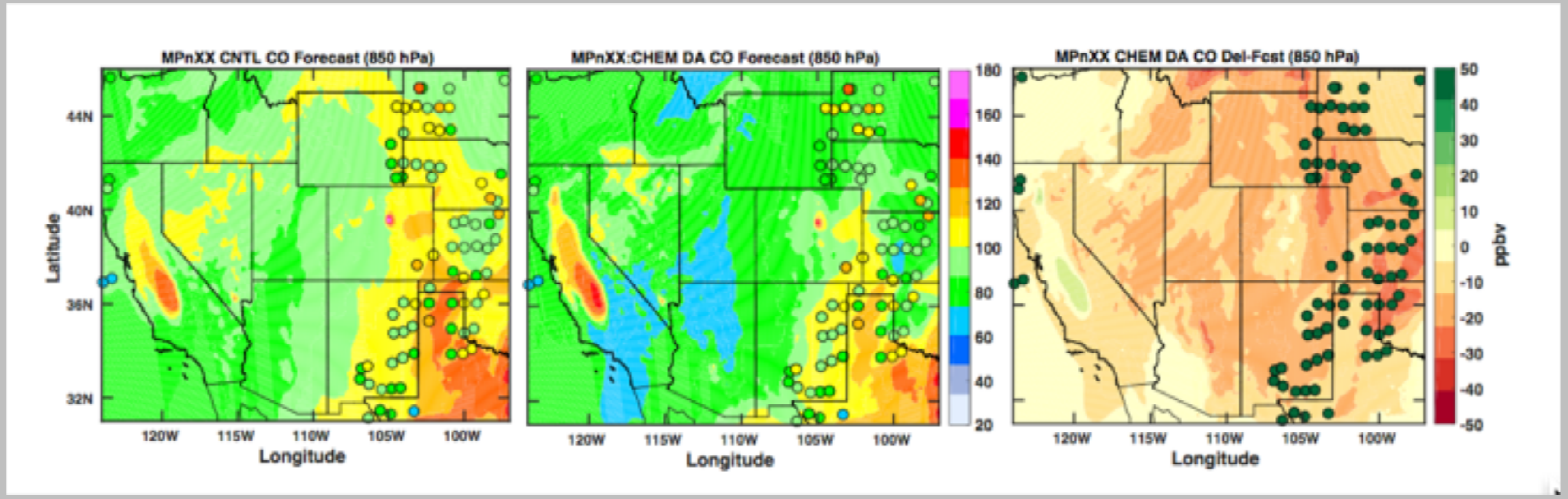
FRAPPE: Forecast Verification



Arthur P. Mizzi
mizzi@ucar.edu

FRAPPE: (MOPIT CO RETRs)

July 21, 2014 18 UTC

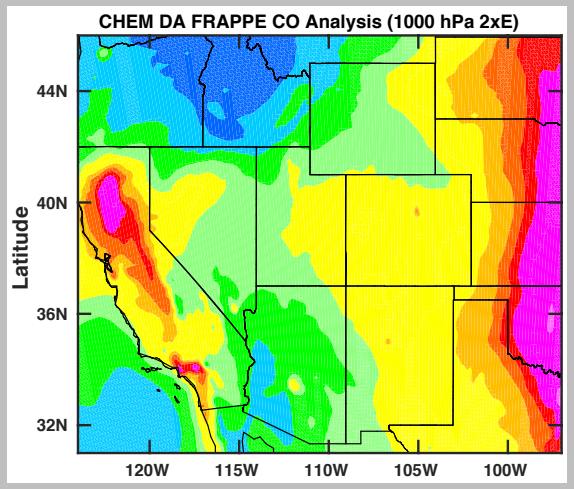


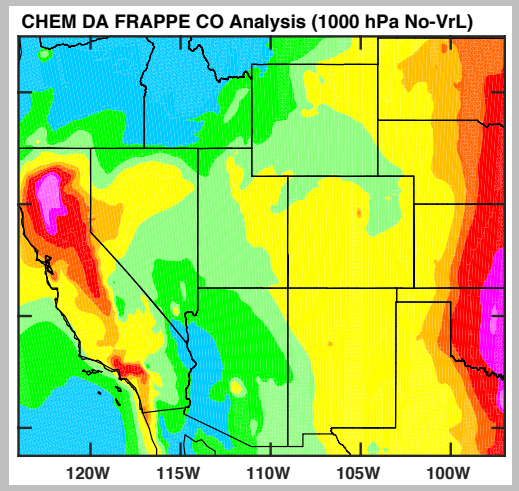
Met Ex **RETR Ex** **Met Ex – RETR Ex**

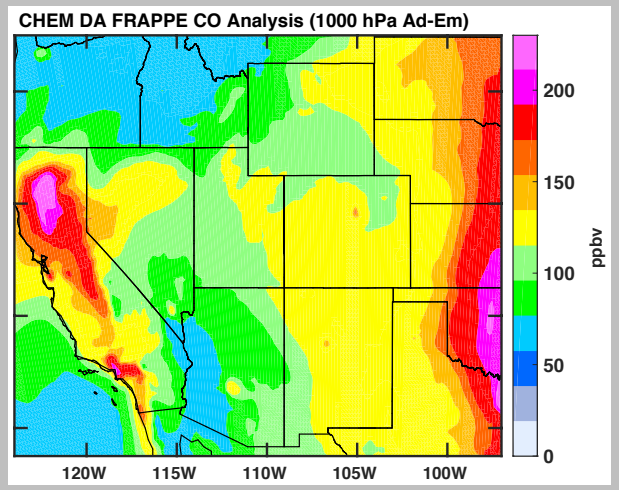
Arthur P. Mizzi
mizzi@ucar.edu

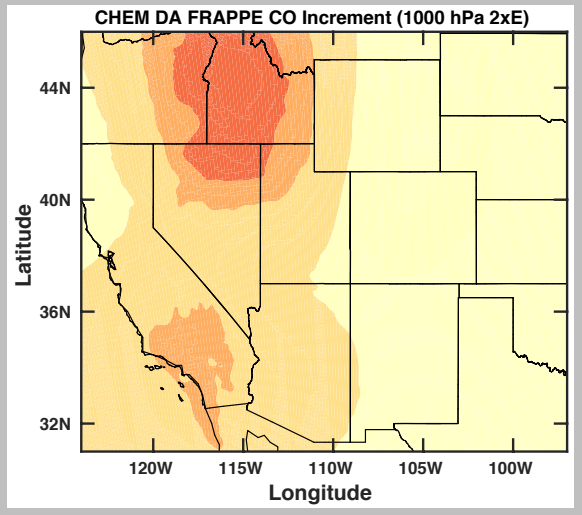
FRAPPE: (MOPIT CO RETRs)

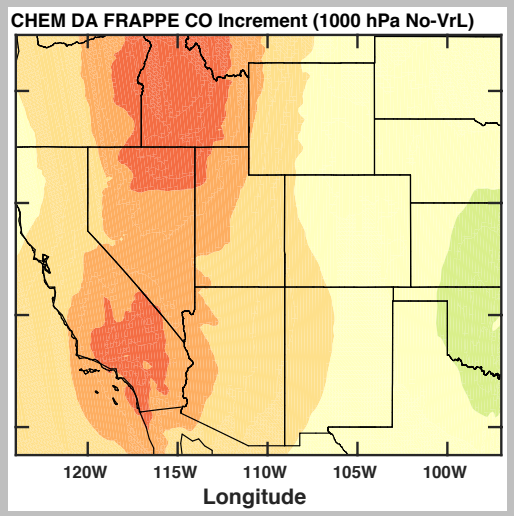
July 20, 2014 18 UTC

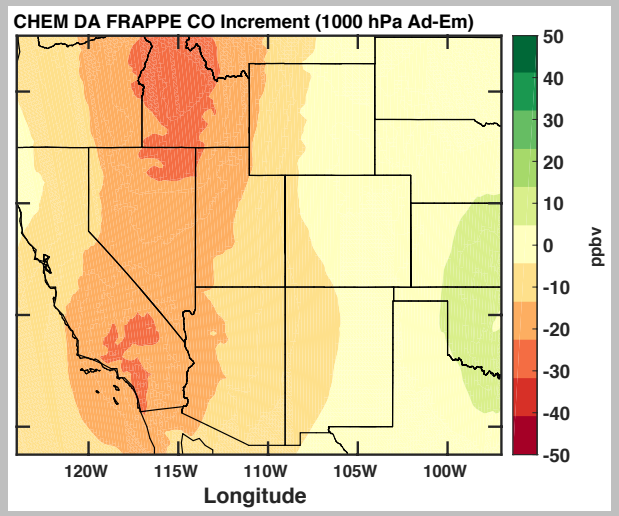












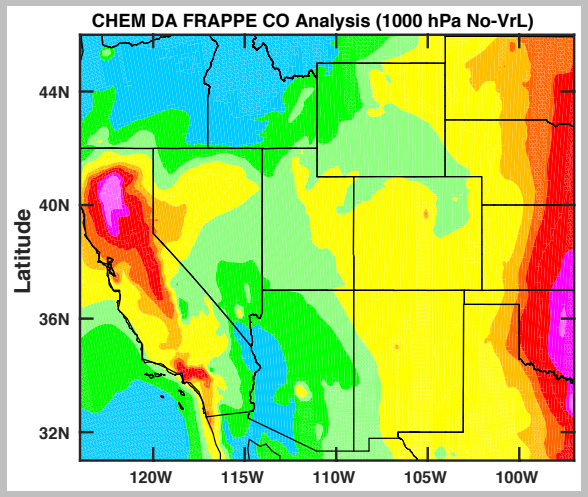
RETR Ex | **No State Variable Loc** | **No Loc and Adj Emiss**

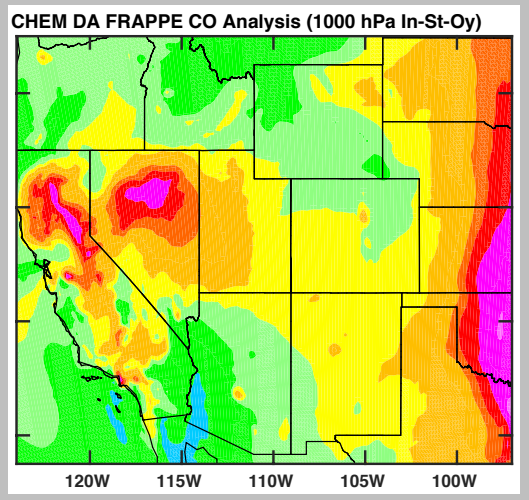
Arthur P. Mizzi
mizzi@ucar.edu

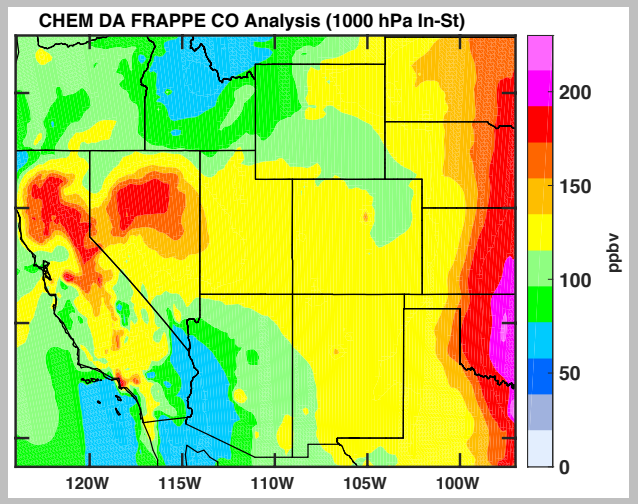
15

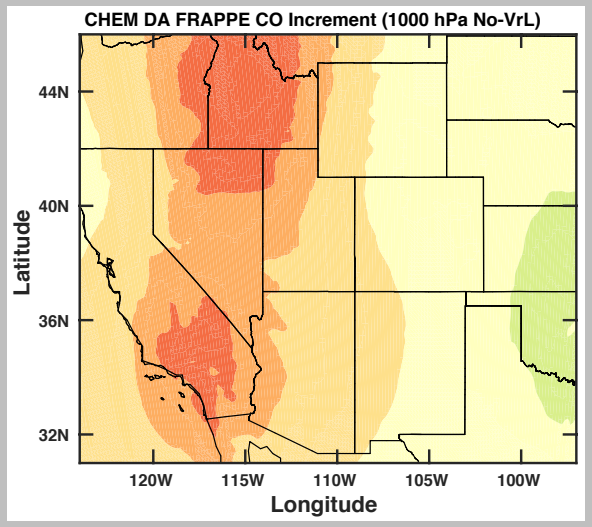
FRAPPE CO (MOPITT CO and AirNow CO, O3)

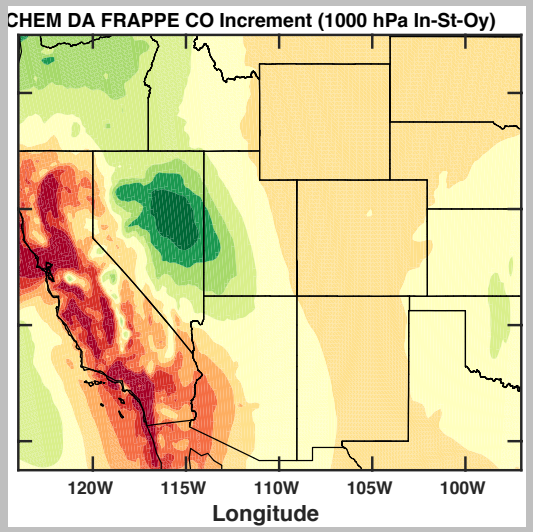
July 20, 2014 18 UTC

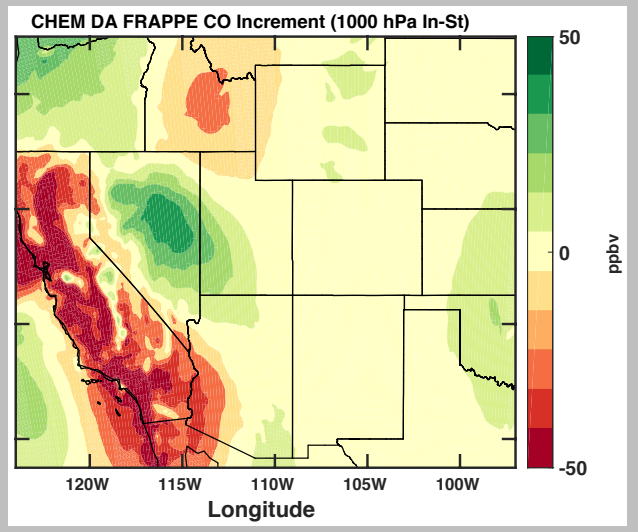










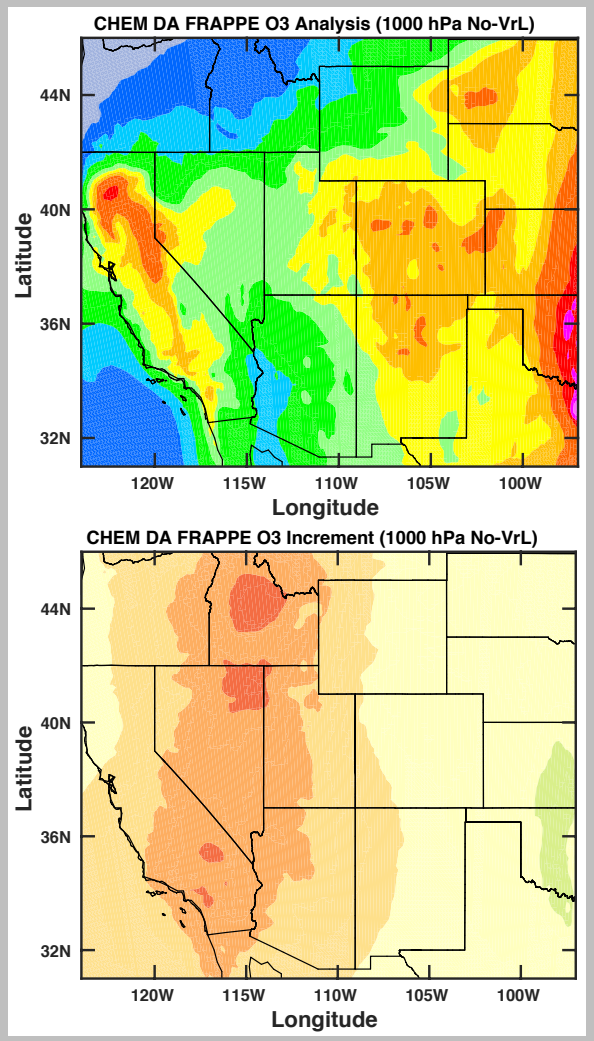


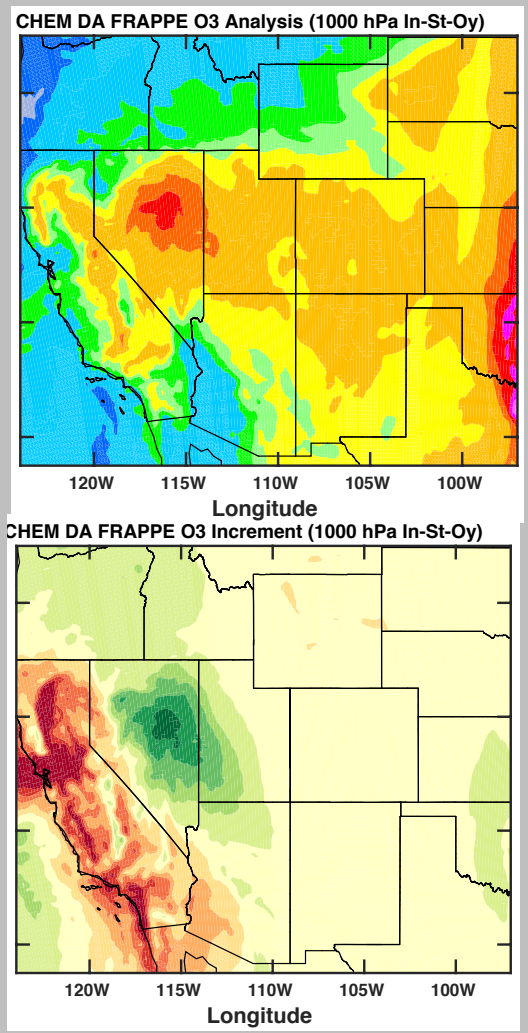
No State Variable Loc | **In Situ** | **No Loc and In Situ**

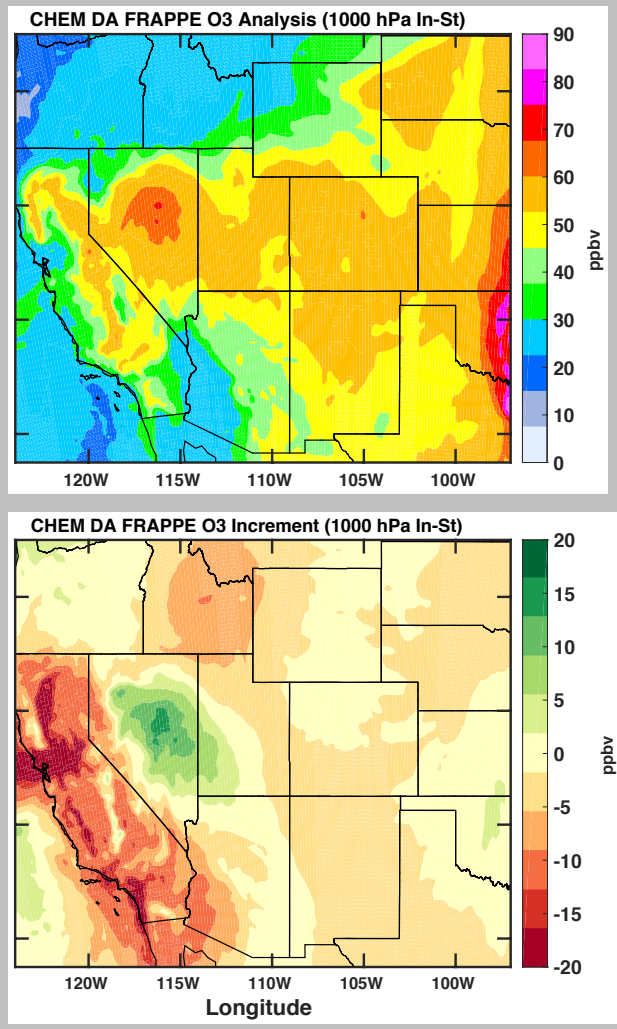
Arthur P. Mizzi
mizzi@ucar.edu

FRAPPE O3 (MOPITT CO and AirNow CO, O3)

July 20, 2014 18 UTC







No State Variable Loc | **In Situ** | **No Loc and In Situ**

Arthur P. Mizzi
mizzi@ucar.edu

Special Challenges with Assimilation of Observations of Atmospheric Composition

Arthur P. Mizzi
mizzi@ucar.edu

Chemical Data Assimilation Challenges

- Sparse observations, mostly surface observations collected for regulatory purposes.
- Upper atmosphere/profile observations are primarily satellite-based retrievals (indirect observations, large data volume, low information density, large storage and assimilation computational requirements).
- Retrievals contain redundant information and influence of retrieval prior through the retrieval equation.
- Interaction of meteorology and chemistry observations on state variables during assimilation can disrupt chemistry (requires state-variable localization).

Arthur P. Mizzi
mizzi@ucar.edu

Chemical Data Assimilation Challenges cont.

- Under-sampling and spread collapse issues require radius-of-influence localization and inflation but appropriate localization radius and spread uncertain.
- Forecast results are sensitive to correct emissions but large uncertainty in emissions and emissions error covariance is unknown.
- Retrieval error covariance products generally contain error cross-correlations.
- Chemical variables and errors tend to have non-Gaussian distributions.
- How to do vertical localization for assimilation of retrieval profiles is uncertain.

Arthur P. Mizzi
mizzi@ucar.edu

Review of WRF-Chem/DART Directory Structure and the Tutorial Exercise on Localization

Arthur P. Mizzi
mizzi@ucar.edu

The DART "../models/wrf_chem" Directory

mizzi on cheyenne5: /glade/p/work/mizzi/TRUNK/DART_CHEM_REPOSITORY/models/wrf_chem

```
total 4096
drwxr-xr-x  2 mizzi ncar    512 Feb 21  2017 dart_clamping/
drwxr-xr-x  2 mizzi ncar 131072 Jan 10 16:19 hybrid_scripts/
drwxr-xr-x  2 mizzi ncar 131072 Feb 21  2017 matlab/
-rw-r--r--  1 mizzi ncar 455895 Jan  9 09:27 model_mod.f90
-rw-r--r--  1 mizzi ncar 455894 Oct 18 15:35 model_mod.f90~
-rw-r--r--  1 mizzi ncar  48385 Feb 21  2017 model_mod.html
-rw-r--r--  1 mizzi ncar   4308 Feb 21  2017 model_mod.nml
-rw-r--r--  1 mizzi ncar  83400 Feb 21  2017 module_map_utils.f90
-rw-r--r--  1 mizzi ncar  73570 Feb 21  2017 module_map_utils-wrf3.0.f90
drwxr-xr-x  4 mizzi ncar    512 Feb 21  2017 namelist_scripts/
drwxr-xr-x  3 mizzi ncar    512 Feb 21  2017 PERTURB/
drwxr-xr-x  6 mizzi ncar    512 Feb 21  2017 regression/
drwxr-xr-x  3 mizzi ncar 131072 Nov 13 17:51 run_diagnostics/
drwxr-xr-x  3 mizzi ncar    512 Feb 21  2017 run_diagnostics_OMI/
drwxr-xr-x  2 mizzi ncar 131072 Feb 21  2017 run_frappe_diagnostics/
drwxr-xr-x 27 mizzi ncar 131072 Oct 18 13:07 run_scripts/
-rw-r--r--  1 mizzi ncar   5504 Feb 21  2017 select.f90
drwxr-xr-x  2 mizzi ncar 131072 Apr 20  2017 shell_scripts/
-rw-r--r--  1 mizzi ncar    106 Jan  9 09:25 svn-commit.tmp~
drwxr-xr-x  2 mizzi ncar 131072 Jan  9 20:18 work/
drwxr-xr-x  2 mizzi ncar    512 Feb 21  2017 WRF_BC/
drwxr-xr-x  2 mizzi ncar 131072 Feb 21  2017 WRFCHEM_BUG_FIXES/
drwxr-xr-x  2 mizzi ncar    512 Feb 21  2017 wrfchem_code_fixes/
drwxr-xr-x  2 mizzi ncar 131072 Oct 26 11:20 WRF_DART_utilities/
-rw-r--r--  1 mizzi ncar   4159 Feb 21  2017 wrf_state_variables_table
cheyenne5:/glade/p/work/mizzi/TRUNK/DART_CHEM_REPOSITORY/models/wrf_chem>
```

hybrid_scripts – various WRF-Chem/DART utilities.
namelist_scripts – scripts to setup the WRF, WRF-Chem, WPS, WRFDA, and DART namelists.
run_diagnostics – scripts to run WRF-Chem/DART diagnostics and plotting,
run_scripts – the main WRF-Chem/DART run scripts directory.

Arthur P. Mizzi
mizzi@ucar.edu

The DART "../models/wrf_chem/run_scripts" Directory

mizzi on cheyenne5: /glade/p/work/mizzi/TRUNK/DART_CHEM_REPOSITORY/models/wrf_chem/run_scripts

```
total 1536
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_AIRNOW/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_AVG_FIX/
drwxr-xr-x 2 mizzi ncar    512 Mar 29  2017 RUN_BAND_DEPTH/
drwxr-xr-x 2 mizzi ncar    512 Sep 20 14:58 RUN_CHINA_EPA/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_CLOSEST_MEMBER/
drwxr-xr-x 2 mizzi ncar 131072 Sep 21 11:32 RUN_EMISS_INV/
drwxr-xr-x 4 mizzi ncar    512 Feb 21  2017 RUN_IASI/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_IASI_CO/
drwxr-xr-x 3 mizzi ncar    512 Feb 21  2017 RUN_IASI_O3/
drwxr-xr-x 9 mizzi ncar    512 Feb 21  2017 RUN_INITIAL_EMISS/
drwxr-xr-x 2 mizzi ncar 131072 Feb 21  2017 RUN_INITIAL_ENS/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_LOCALIZE_OBS_SEQ/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_MODIS/
drwxr-xr-x 4 mizzi ncar    512 Feb 21  2017 RUN_MOPITT/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_MOPITT_CO/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_NCEP/
drwxr-xr-x 2 mizzi ncar 131072 Feb 21  2017 RUN_OBS_SEQ_TOOL/
drwxr-xr-x 3 mizzi ncar    512 Feb 21  2017 RUN_OMI_TEST/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_PERFECT_MODEL/
drwxr-xr-x 4 mizzi ncar    512 Feb 22  2017 RUN_PERT_CHEM/
drwxr-xr-x 2 mizzi ncar    512 Feb 21  2017 RUN_PREPROC/
drwxr-xr-x 3 mizzi ncar 131072 Jan 11 15:06 RUN_REAL_TIME/
drwxr-xr-x 2 mizzi ncar 131072 Jan 10 14:51 RUN_REAL_TIME_CHEYENNE/
drwxr-xr-x 2 mizzi ncar    512 Nov 14 09:20 RUN_TIME_INTERP/
drwxr-xr-x 9 mizzi ncar 131072 Feb 21  2017 RUN_WRF_CHEM/
cheyenne5:/glade/p/work/mizzi/TRUNK/DART_CHEM_REPOSITORY/models/wrf_chem/run_scripts>
```

RUN_REAL_TIME – the WRF-Chem/DART real time run scripts directory. This is where you will find the scripts for running WRF-Chem/DART. The "real_time_PANDA_RETR_RELEASE_TEST.ksh" script runs the released test case.

RUN_REAL_TIME_CHEYENNE – a development directory for converting the real time runs scripts to Cheyenne. Eventually, this directory will disappear and the run scripts for Cheyenne will be in RUN_REAL_TIME.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/ksh -aeux
#################################################################
#
# Purpose: Run_the WRF-Chem/DART PANDA test case
#
# NOTE: To generate chemistry perturbations check NL_SW_GENERATE comments
# To use the same purturbastions from one cycle to the next one must copy
# the pert_file_emiss and pert_file_icbc file to the run directory
#
#################################################################
#
# CYCLE DATE-TIME:
export CYCLE_STR_DATE=2014071400
# initial cycle
#export CYCLE_STR_DATE=2014072418
# first assimilation cycle
export CYCLE_STR_DATE=2014072500
#
export CYCLE_END_DATE=${CYCLE_STR_DATE}
export CYCLE_END_DATE=2014072500
export CYCLE_DATE=${CYCLE_STR_DATE}
export NL_FAC_OBS_ERROR_MOPITT=1.00
export NL_FAC_OBS_ERROR_IASI=1.00
export RETRIEVAL_TYPE_MOPITT=RETR
export RETRIEVAL_TYPE_IASI=RAWR
#
export USE_LOG_CO=false
if [[ ${USE_LOG_CO} == true ]]; then
   export CO_MIN=NULL
   export CO_MAX=NULL
   export USE_LOG_CO_LOGIC=.true.
else
   export CO_MIN=1.e-4
   export CO_MAX=NULL
   export USE_LOG_CO_LOGIC=.false.
fi
#
# If VARLOC = true, then INDEP_CHEM_ASIM = false
# If INDEP_CHEM_ASIM = true, then VARLOC = false
# VARLOC and INDEP_CHEM_ASIM may both be false but
# they may not both be true
#
export VARLOC=.true.
export INDEP_CHEM_ASIM=.false.
#
export ADD_EMISS=.false.
export EMISS_DAMP_CYCLE=0.5
export EMISS_DAMP_INTRA_CYCLE=0.5
#
let BAND_ISO_VAL_CO=.09
#
```

Set the cycle start and end date/times.

Set some experiment parameters:
NL_FAC_OBS_ERROR_xxx – the scaling for the observation error of the xxx retrievals.
RETRIEVAL_TYPE_xxx – the retrieval assimilation form:
RAWR – mixing ratio, RETR – log10 of mixing ratio,
QOR – quasi-optimal, and CPSR – compact phase space retrievals.

USE_LOG_CO – switch to do assimilation in log space.

VARLOC – switch to use state variable localization.
INDEP_CHEM_ASSIM – switch to use independent chemistry assimilation.
ADD_EMISS – switch to use emission inversion
EMISS_DAMP_CYCLE – cycle time damping factor for emissions adjustment.
EMISS_DAMP_INTRA - intra-cycle time damping factor for emissions adjustment.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
#################################################################
# Run fine scale forecast only
export RUN_FINE_SCALE=false
#
# Restart fine scale forecast only
export RUN_FINE_SCALE_RESTART=false
export RESTART_DATE=2014072312
#
if [[ ${RUN_FINE_SCALE_RESTART} = "true" ]]; then
   export RUN_FINE_SCALE=true
fi
#
# Run WRF-Chem for failed forecasts
export RUN_SPECIAL_FORECAST=false
export NUM_SPECIAL_FORECAST=1
export SPECIAL_FORECAST_FAC=1.
export SPECIAL_FORECAST_FAC=1./2.
export SPECIAL_FORECAST_FAC=2./3.
export SPECIAL_FORECAST_MEM[1]=4
export SPECIAL_FORECAST_MEM[2]=2
export SPECIAL_FORECAST_MEM[3]=3
export SPECIAL_FORECAST_MEM[4]=4
export SPECIAL_FORECAST_MEM[5]=5
export SPECIAL_FORECAST_MEM[6]=6
export SPECIAL_FORECAST_MEM[7]=7
export SPECIAL_FORECAST_MEM[8]=8
export SPECIAL_FORECAST_MEM[9]=9
export SPECIAL_FORECAST_MEM[10]=10
#
# Run temporal interpolation for missing background files
export RUN_INTERPOLATE=false
#
# for 2014072212 and 2014072218
#export BACK_DATE=2014072206
#export FORW_DATE=2014072300
#let BACK_WT=.3333
#let BACK_WT=.6667
#
# for 20142900
#export BACK_DATE=2014072818
#export FORW_DATE=2014072906
#let BACK_WT=.5000
#
# for 20142912
export BACK_DATE=2014072906
export FORW_DATE=2014072918
let BACK_WT=.5000
#
```

RUN_FINE_SCALE – switch to run fine grid forecast.
RUN_FINE_SCALE_RESTART – switch to restart fine grid forecast.
REESTART_DATE – Date/time to restart fine grid forecast.

RUN_SPECIAL_FORECAST - switch to rerun failed ensemble members.
NUM_SPECIAL_FORECAST - number of ensemble forecasts to rerun.
SPECIAL_FORECAST_FAC – time step scaling factor for the ensemble forecast reruns.
SPECICAL_FORECAST_MEM[ ] – the member id for the ensemble forecast reruns.

Code to enable temporal interpolation of the large scale meteorology forecast fields.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
#########################################################################
while [[ ${CYCLE_DATE} -le ${CYCLE_END_DATE} ]]; do
export DATE=${CYCLE_DATE}
export INITIAL_DATE=2014072418
export FIRST_FILTER_DATE=2014072500
export FIRST_EMISS_INV_DATE=2014072500
export FIRST_DART_INFLATE_DATE=2014072500
if [[ ${ADD_EMISS} == false ]]; then
   export FIRST_EMISS_INV_DATE=${DATE}
fi
export CYCLE_PERIOD=6
export HISTORY_INTERVAL_HR=1
(( HISTORY_INTERVAL_MIN = ${HISTORY_INTERVAL_HR} * 60 ))
export START_IASI_O3_DATA=2014060100
export END_IASI_O3_DATA=2014073118
export NL_DEBUG_LEVEL=200
#
# CODE VERSIONS:
export WPS_VER=WPSv3.6.1_dmpar
export WPS_GEOG_VER=WPSv3.6.1_GEOG_DATA
export WRFDA_VER=WRFDAv3.4_dmpar
export WRF_VER=WRFv3.6.1_dmpar
export WRFCHEM_VER=WRFCHEMv3.6.1_dmpar
export DART_VER=DART_CHEM_REPOSITORY
#
# ROOT DIRECTORIES:
export SCRATCH_DIR=/glade/scratch/mizzi
export WORK_DIR=/glade/p/work/mizzi
export ACD_DIR=/glade/p/acd/mizzi
export FRAPPE_DIR=/glade/p/FRAPPE
#
# DEPENDENT INPUT DATA DIRECTORIES:
export EXPERIMENT_DIR=${FRAPPE_DIR}
export RUN_DIR=${EXPERIMENT_DIR}/real_PANDA_RETR_VARLOC_RELEASE_TEST
export TRUNK_DIR=${WORK_DIR}/TRUNK
export WPS_DIR_v3p4p1=/glade/p/work/wrfhelp/PRE_COMPILED_CODE/WPSV3.4.1_intel_dmpar
export WPS_DIR_v3p7p1=/glade/p/work/wrfhelp/PRE_COMPILED_CODE/WPSV3.7.1_intel_serial_large-file
export WPS_DIR_v3p8p1=/glade/p/work/wrfhelp/PRE_COMPILED_CODE/WPSV3.8.1_intel_serial_large-file
export WPS_DIR=${TRUNK_DIR}/${WPS_VER}
export WPS_GEOG_DIR=${TRUNK_DIR}/${WPS_GEOG_VER}/geog
export WRFCHEM_DIR=${TRUNK_DIR}/${WRFCHEM_VER}
export WRFDA_DIR=${TRUNK_DIR}/${WRFDA_VER}
export DART_DIR=${TRUNK_DIR}/${DART_VER}
export BUILD_DIR=${WRFDA_DIR}/var/da
export WRF_DIR=${TRUNK_DIR}/${WRF_VER}
export HYBRID_SCRIPTS_DIR=${DART_DIR}/models/wrf_chem/hybrid_scripts
export ADJUST_EMISS_DIR=${DART_DIR}/models/wrf_chem/run_scripts/RUN_EMISS_INV
export EXPERIMENT_DATA_DIR=${EXPERIMENT_DIR}/PANDA_REAL_TIME_DATA
#export EXPERIMENT_DATA_DIR=${EXPERIMENT_DIR}/PANDA_REAL_TIME_DATA_IDIR
```

INITIAL_DATE, FIRST_FILTER_DATE, FIRST_EMISS_INV_DATE, FIRST_DART_INFLATE_DATE – these parameters identify the cycle on which the respective task starts.

Set cycling and assimilation parameters.

Set the respective code versions.

Set the respective independent directory paths.

Set the respective dependent directory paths.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
export MOZBC_DATA_DIR=${EXPERIMENT_DATA_DIR}/mozart_forecasts
export EXPERIMENT_STATIC_FILES=${EXPERIMENT_DATA_DIR}/static_files
export EXPERIMENT_WRFCHEMI_DIR=${EXPERIMENT_DATA_DIR}/anthro_emissions
export EXPERIMENT_WRFFIRECHEMI_DIR=${EXPERIMENT_DATA_DIR}/fire_emissions
export EXPERIMENT_WRFBIOCHEMI_DIR=${EXPERIMENT_DATA_DIR}/bio_emissions
export EXPERIMENT_COLDENS_DIR=${EXPERIMENT_DATA_DIR}/wes_coldens
export EXPERIMENT_PREPBUFR_DIR=${EXPERIMENT_DATA_DIR}/met_obs_prep_data
export EXPERIMENT_MOPITT_CO_DIR=${EXPERIMENT_DATA_DIR}/mopitt_co_hdf_data
export EXPERIMENT_IASI_CO_DIR=${EXPERIMENT_DATA_DIR}/iasi_co_hdf_data
export EXPERIMENT_IASI_O3_DIR=${EXPERIMENT_DATA_DIR}/iasi_o3_hdf_data
export EXPERIMENT_AIRNOW_DIR=${EXPERIMENT_DATA_DIR}/airnow_csv_data
export EXPERIMENT_PANDA_DIR=${EXPERIMENT_DATA_DIR}/panda_csv_data
export EXPERIMENT_MODIS_AOD_DIR=${EXPERIMENT_DATA_DIR}/modis_aod_hdf_data
export EXPERIMENT_GFS_DIR=${EXPERIMENT_DATA_DIR}/gfs_forecasts
export EXPERIMENT_DUST_DIR=${EXPERIMENT_DATA_DIR}/dust_fields
export EXPERIMENT_HIST_IO_DIR=${EXPERIMENT_DATA_DIR}/hist_io_files
export VTABLE_DIR=${WPS_DIR}/ungrib/Variable_Tables
#export VTABLE_DIR=${WPS_DIR_v3p7p1}/ungrib/Variable_Tables
export BE_DIR=${WRFDA_DIR}/var/run
export PERT_CHEM_INPUT_DIR=${DART_DIR}/models/wrf_chem/run_scripts/RUN_PERT_CHEM/ICBC_PERT
export PERT_CHEM_EMISS_DIR=${DART_DIR}/models/wrf_chem/run_scripts/RUN_PERT_CHEM/EMISS_PERT
export RUN_BAND_DEPTH_DIR=${DART_DIR}/models/wrf_chem/run_scripts/RUN_BAND_DEPTH
#
cp ${DART_DIR}/models/wrf_chem/work/advance_time ./.
cp ${DART_DIR}/models/wrf_chem/work/input.nml ./.
export YYYY=$(echo $DATE | cut -c1-4)
export YY=$(echo $DATE | cut -c3-4)
export MM=$(echo $DATE | cut -c5-6)
export DD=$(echo $DATE | cut -c7-8)
export HH=$(echo $DATE | cut -c9-10)
export DATE_SHORT=${YY}${MM}${DD}${HH}
export FILE_DATE=${YYYY}-${MM}-${DD}_${HH}:00:00
export PAST_DATE=$(${BUILD_DIR}/da_advance_time.exe ${DATE} -${CYCLE_PERIOD} 2>/dev/null)
export PAST_YYYY=$(echo $PAST_DATE | cut -c1-4)
export PAST_YY=$(echo $PAST_DATE | cut -c3-4)
export PAST_MM=$(echo $PAST_DATE | cut -c5-6)
export PAST_DD=$(echo $PAST_DATE | cut -c7-8)
export PAST_HH=$(echo $PAST_DATE | cut -c9-10)
export PAST_FILE_DATE=${PAST_YYYY}-${PAST_MM}-${PAST_DD}_${PAST_HH}:00:00
export NEXT_DATE=$(${BUILD_DIR}/da_advance_time.exe ${DATE} +${CYCLE_PERIOD} 2>/dev/null)
export NEXT_YYYY=$(echo $NEXT_DATE | cut -c1-4)
export NEXT_YY=$(echo $NEXT_DATE | cut -c3-4)
export NEXT_MM=$(echo $NEXT_DATE | cut -c5-6)
export NEXT_DD=$(echo $NEXT_DATE | cut -c7-8)
export NEXT_HH=$(echo $NEXT_DATE | cut -c9-10)
export NEXT_FILE_DATE=${NEXT_YYYY}-${NEXT_MM}-${NEXT_DD}_${NEXT_HH}:00:00
#
```

Continue setting the respective dependent directory paths.

Define date/time parameters for this cycle.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
#
# DART TIME DATA
export DT_YYYY=${YYYY}
export DT_YY=${YY}
export DT_MM=${MM}
export DT_DD=${DD}
export DT_HH=${HH}
(( DT_MM = ${DT_MM} + 0 ))
(( DT_DD = ${DT_DD} + 0 ))
(( DT_HH = ${DT_HH} + 0 ))
if [[ ${HH} -eq 0 ]]; then
   export TMP_DATE=$(${BUILD_DIR}/da_advance_time.exe ${DATE} -1 2>/dev/null)
   export TMP_YYYY=$(echo $TMP_DATE | cut -c1-4)
   export TMP_YY=$(echo $TMP_DATE | cut -c3-4)
   export TMP_MM=$(echo $TMP_DATE | cut -c5-6)
   export TMP_DD=$(echo $TMP_DATE | cut -c7-8)
   export TMP_HH=$(echo $TMP_DATE | cut -c9-10)
   export D_YYYY=${TMP_YYYY}
   export D_YY=${TMP_YY}
   export D_MM=${TMP_MM}
   export D_DD=${TMP_DD}
   export D_HH=24
   (( DD_MM = ${D_MM} + 0 ))
   (( DD_DD = ${D_DD} + 0 ))
   (( DD_HH = ${D_HH} + 0 ))
else
   export D_YYYY=${YYYY}
   export D_YY=${YY}
   export D_MM=${MM}
   export D_DD=${DD}
   export D_HH=${HH}
   (( DD_MM = ${D_MM} + 0 ))
   (( DD_DD = ${D_DD} + 0 ))
   (( DD_HH = ${D_HH} + 0 ))
fi
export D_DATE=${D_YYYY}${D_MM}${D_DD}${D_HH}
#
# CALCULATE GREGORIAN TIMES FOR START AND END OF ASSIMILATION WINDOW
set -A GREG_DATA `echo $DATE 0 -g | ${DART_DIR}/models/wrf_chem/work/advance_time`
export DAY_GREG=${GREG_DATA[0]}
export SEC_GREG=${GREG_DATA[1]}
set -A GREG_DATA `echo $NEXT_DATE 0 -g | ${DART_DIR}/models/wrf_chem/work/advance_time`
export NEXT_DAY_GREG=${GREG_DATA[0]}
export NEXT_SEC_GREG=${GREG_DATA[1]}
export ASIM_WINDOW=3
export ASIM_MIN_DATE=$($BUILD_DIR/da_advance_time.exe $DATE -$ASIM_WINDOW 2>/dev/null)
export ASIM_MIN_YYYY=$(echo $ASIM_MIN_DATE | cut -c1-4)
export ASIM_MIN_YY=$(echo $ASIM_MIN_DATE | cut -c3-4)
export ASIM_MIN_MM=$(echo $ASIM_MIN_DATE | cut -c5-6)
export ASIM_MIN_DD=$(echo $ASIM_MIN_DATE | cut -c7-8)
export ASIM_MIN_HH=$(echo $ASIM_MIN_DATE | cut -c9-10)
```

Continue setting date/time parameters for this cycle.

Set assimilation window parameters.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
export ASIM_MAX_DATE=$($BUILD_DIR/da_advance_time.exe $DATE +$ASIM_WINDOW 2>/dev/null)
export ASIM_MAX_YYYY=$(echo $ASIM_MAX_DATE | cut -c1-4)
export ASIM_MAX_YY=$(echo $ASIM_MAX_DATE | cut -c3-4)
export ASIM_MAX_MM=$(echo $ASIM_MAX_DATE | cut -c5-6)
export ASIM_MAX_DD=$(echo $ASIM_MAX_DATE | cut -c7-8)
export ASIM_MAX_HH=$(echo $ASIM_MAX_DATE | cut -c9-10)
set -A temp `echo $ASIM_MIN_DATE 0 -g | ${DART_DIR}/models/wrf_chem/work/advance_time`
export ASIM_MIN_DAY_GREG=${temp[0]}
export ASIM_MIN_SEC_GREG=${temp[1]}
set -A temp `echo $ASIM_MAX_DATE 0 -g | ${DART_DIR}/models/wrf_chem/work/advance_time`
export ASIM_MAX_DAY_GREG=${temp[0]}
export ASIM_MAX_SEC_GREG=${temp[1]}
#
# SELECT COMPONENT RUN OPTIONS:
if [[ ${RUN_SPECIAL_FORECAST} = "false" ]]; then
   export RUN_GEOGRID=false
   export RUN_UNGRIB=true
   export RUN_METGRID=true
   export RUN_REAL=true
   export RUN_PERT_WRFCHEM_MET_IC=true
   export RUN_PERT_WRFCHEM_MET_BC=true
   export RUN_EXO_COLDENS=true
   export RUN_SEASON_WES=true
   export RUN_WRFCHEM_BIO=true
   export RUN_WRFCHEM_FIRE=true
   export RUN_WRFCHEM_CHEMI=true
   export RUN_PERT_WRFCHEM_CHEM_ICBC=true
   export RUN_PERT_WRFCHEM_CHEM_EMISS=true
   export RUN_MOPITT_CO_OBS=true
   export RUN_IASI_CO_OBS=false
   export RUN_IASI_O3_OBS=false
   export RUN_AIRNOW_O3_OBS=false
   export RUN_AIRNOW_CO_OBS=false
   export RUN_PANDA_CO_OBS=false
   export RUN_PANDA_O3_OBS=false
   export RUN_PANDA_PM25_OBS=false
   export RUN_MODIS_AOD_OBS=false
   export RUN_MET_OBS=true
   export RUN_COMBINE_OBS=true
   export RUN_PREPROCESS_OBS=true
#
   if [[ ${DATE} -eq ${INITIAL_DATE}  ]]; then
      export RUN_WRFCHEM_INITIAL=true
      export RUN_DART_FILTER=false
      export RUN_UPDATE_BC=false
      export RUN_WRFCHEM_CYCLE_CR=false
      export RUN_BAND_DEPTH=false
      export RUN_WRFCHEM_CYCLE_FR=false
      export RUN_ENSEMBLE_MEAN_INPUT=false
      export RUN_ENSMEAN_CYCLE_FR=false
      export RUN_ENSEMBLE_MEAN_OUTPUT=false
```

Continue setting assimilation window parameters.

Set flags to control task execution. Currently, these tasks are run sequentially and ordered similarly to the order of these flags. This is the only section where these flags should be set by the user. Subsequent flag setting sections adjust the user flag settings for special cycles.

Flag settings for the initial cycle. This section should not be changed by the user.

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
# FORECAST PARAMETERS:
export USE_DART_INFL=true
export FCST_PERIOD=6
(( CYCLE_PERIOD_SEC=${CYCLE_PERIOD}*60*60 ))
export NUM_MEMBERS=10
export MAX_DOMAINS=02
export CR_DOMAIN=01
export FR_DOMAIN=02
export NNXP_CR=149
export NNYP_CR=111
export NNZP_CR=36
export NNXP_FR=147
export NNYP_FR=156
export NNZP_FR=36
(( NNXP_STAG_CR=${NNXP_CR}+1 ))
(( NNYP_STAG_CR=${NNYP_CR}+1 ))
(( NNZP_STAG_CR=${NNZP_CR}+1 ))
(( NNXP_STAG_FR=${NNXP_FR}+1 ))
(( NNYP_STAG_FR=${NNYP_FR}+1 ))
(( NNZP_STAG_FR=${NNZP_FR}+1 ))
export NNZ_CHEM=10
export NNCHEM_SPC=24
export NNFIRE_SPC=31
export NNBIO_SPC=1
export NZ_CHEMI=${NNZ_CHEM}
export NZ_FIRECHEMI=1
export NCHEMI_EMISS=2
export NFIRECHEMI_EMISS=7
export ISTR_CR=1
export JSTR_CR=1
export ISTR_FR=59
export JSTR_FR=44
export DX_CR=60000
export DX_FR=20000
(( LBC_END=2*${FCST_PERIOD} ))
export LBC_FREQ=3
(( INTERVAL_SECONDS=${LBC_FREQ}*60*60 ))
export LBC_START=0
export START_DATE=${DATE}
export END_DATE=$($BUILD_DIR/da_advance_time.exe ${START_DATE} ${FCST_PERIOD} 2>/dev/null)
export START_YEAR=$(echo $START_DATE | cut -c1-4)
export START_YEAR_SHORT=$(echo $START_DATE | cut -c3-4)
export START_MONTH=$(echo $START_DATE | cut -c5-6)
export START_DAY=$(echo $START_DATE | cut -c7-8)
export START_HOUR=$(echo $START_DATE | cut -c9-10)
export START_FILE_DATE=${START_YEAR}-${START_MONTH}-${START_DAY}_${START_HOUR}:00:0
export END_YEAR=$(echo $END_DATE | cut -c1-4)
export END_MONTH=$(echo $END_DATE | cut -c5-6)
export END_DAY=$(echo $END_DATE | cut -c7-8)
export END_HOUR=$(echo $END_DATE | cut -c9-10)
export END_FILE_DATE=${END_YEAR}-${END_MONTH}-${END_DAY}_${END_HOUR}:00:00
#
```

Forecast model grid parameter settings.

Forecast model time parameter settings.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
# LARGE SCALE FORECAST PARAMETERS:
export FG_TYPE=GFS
export GRIB_PART1=gfs_4_
export GRIB_PART2=.g2.tar
#
# COMPUTER PARAMETERS:
export PROJ_NUMBER=NACD0002
export PROJ_NUMBER_ACD=P19010000
export PROJ_NUMBER_NSC=NACD0006
export PROJ_NUMBER_NSC=${PROJ_NUMBER_ACD}
export GENERAL_JOB_CLASS=geyser
export GEOGRID_TIME_LIMIT=0:10
export GEOGRID_NUM_TASKS=32
export GEOGRID_TASKS_PER_NODE=8
export GEOGRID_JOB_CLASS=geyser
export WRFCHEM_TIME_LIMIT=6:00
export WRFCHEM_NUM_TASKS=32
export WRFCHEM_TASKS_PER_NODE=8
export WRFCHEM_JOB_CLASS=geyser
export WRFDA_TIME_LIMIT=0:20
export WRFDA_NUM_TASKS=32
export WRFDA_TASKS_PER_NODE=8
export WRFDA_JOB_CLASS=geyser
export FILTER_TIME_LIMIT=7:59
export FILTER_NUM_TASKS=32
export FILTER_TASKS_PER_NODE=8
export FILTER_JOB_CLASS=geyser
export MISC_TIME_LIMIT=0:02
export MISC_NUM_TASKS=1
export MISC_TASKS_PER_NODE=8
export MISC_JOB_CLASS=geyser
#
# RUN DIRECTORIES
export GEOGRID_DIR=${RUN_DIR}/geogrid
export METGRID_DIR=${RUN_DIR}/${DATE}/metgrid
export REAL_DIR=${RUN_DIR}/${DATE}/real
export WRFCHEM_MET_IC_DIR=${RUN_DIR}/${DATE}/wrfchem_met_ic
export WRFCHEM_MET_BC_DIR=${RUN_DIR}/${DATE}/wrfchem_met_bc
export EXO_COLDENS_DIR=${RUN_DIR}/${DATE}/exo_coldens
export SEASONS_WES_DIR=${RUN_DIR}/${DATE}/seasons_wes
export WRFCHEM_BIO_DIR=${RUN_DIR}/${DATE}/wrfchem_bio
export WRFCHEM_FIRE_DIR=${RUN_DIR}/${DATE}/wrfchem_fire
export WRFCHEM_CHEMI_DIR=${RUN_DIR}/${DATE}/wrfchem_chemi
export WRFCHEM_CHEM_EMISS_DIR=${RUN_DIR}/${DATE}/wrfchem_chem_emiss
export WRFCHEM_INITIAL_DIR=${RUN_DIR}/${INITIAL_DATE}/wrfchem_initial
export WRFCHEM_CYCLE_CR_DIR=${RUN_DIR}/${DATE}/wrfchem_cycle_cr
export WRFCHEM_CYCLE_FR_DIR=${RUN_DIR}/${DATE}/wrfchem_cycle_fr
export WRFCHEM_LAST_CYCLE_CR_DIR=${RUN_DIR}/${PAST_DATE}/wrfchem_cycle_cr
export PREPBUFR_MET_OBS_DIR=${RUN_DIR}/${DATE}/prepbufr_met_obs
export MOPITT_CO_OBS_DIR=${RUN_DIR}/${DATE}/mopitt_co_obs
export IASI_CO_OBS_DIR=${RUN_DIR}/${DATE}/iasi_co_obs
export IASI_O3_OBS_DIR=${RUN_DIR}/${DATE}/iasi_o3_obs
```

Computer job submittal parameter settings.

Task run directory path parameter settings. These should not be changed by the user.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
export AIRNOW_CO_OBS_DIR=${RUN_DIR}/${DATE}/airnow_co_obs
export AIRNOW_O3_OBS_DIR=${RUN_DIR}/${DATE}/airnow_o3_obs
export PANDA_CO_OBS_DIR=${RUN_DIR}/${DATE}/panda_co_obs
export PANDA_O3_OBS_DIR=${RUN_DIR}/${DATE}/panda_o3_obs
export PANDA_PM25_OBS_DIR=${RUN_DIR}/${DATE}/panda_pm25_obs
export MODIS_AOD_OBS_DIR=${RUN_DIR}/${DATE}/modis_aod_obs
export COMBINE_OBS_DIR=${RUN_DIR}/${DATE}/combine_obs
export PREPROCESS_OBS_DIR=${RUN_DIR}/${DATE}/preprocess_obs
export WRFCHEM_CHEM_ICBC_DIR=${RUN_DIR}/${DATE}/wrfchem_chem_icbc
export DART_FILTER_DIR=${RUN_DIR}/${DATE}/dart_filter
export UPDATE_BC_DIR=${RUN_DIR}/${DATE}/update_bc
export BAND_DEPTH_DIR=${RUN_DIR}/${DATE}/band_depth
export ENSEMBLE_MEAN_INPUT_DIR=${RUN_DIR}/${DATE}/ensemble_mean_input
export ENSEMBLE_MEAN_OUTPUT_DIR=${RUN_DIR}/${DATE}/ensemble_mean_output
export REAL_TIME_DIR=${DART_DIR}/models/wrf_chem/run_scripts/RUN_REAL_TIME
#
# WPS PARAMETERS:
export SINGLE_FILE=false
export HOR_SCALE=1500
export VTABLE_TYPE=GFS
export METGRID_TABLE_TYPE=ARW
#
# WRF PREPROCESS PARAMETERS
# TARG_LAT=38.92 (91.61)
# TARG_LON=-111.52 = 248.48 (91.61)
#export NL_MIN_LAT=38.32
#export NL_MAX_LAT=39.52
#export NL_MIN_LON=248.08
#export NL_MAX_LON=248.88
#
export NL_MIN_LAT=-15.
export NL_MAX_LAT=60.
export NL_MIN_LON=40.
export NL_MAX_LON=170.
export NNL_MIN_LAT=${NL_MIN_LAT}
export NNL_MAX_LAT=${NL_MAX_LAT}
export NNL_MIN_LON=${NL_MIN_LON}
if [[ ${NL_MIN_LON} -gt 180. ]]; then ((NNL_MIN_LON=${NL_MIN_LON}-360.)) ; fi
export NNL_MAX_LON=${NL_MAX_LON}
if [[ ${NL_MAX_LON} -gt 180. ]]; then ((NNL_MAX_LON=${NL_MAX_LON}-360.)) ; fi
export NL_OBS_PRESSURE_TOP=10000.
#
# PERT CHEM PARAMETERS
export SPREAD_FAC=0.30
export MOZ_SPREAD=${SPREAD_FAC}
export NL_MEAN=1.0
export NL_SPREAD=${SPREAD_FAC}
#
#########################################################################
#
#  NAMELIST PARAMETERS
#
```

Continue setting the task run directory path parameters. These should not be changed by the user.

Set WPS and WRF_DART_Preprocessor parameters.

Set parameters controlling generation of the chemistry state variables and emissions ensembles.

Arthur P. Mizzi
mizzi@ucar.edu

The "real_time_PANDA_RETR_RELEASE_TEST.ksh" Run Script

```
#############################################################################
#
# RUN GEOGRID
#
#############################################################################
#
if [[ ${RUN_GEOGRID} = "true" ]] then
   mkdir -p ${RUN_DIR}/geogrid
   cd ${RUN_DIR}/geogrid
#
   cp ${WPS_DIR}/geogrid.exe ./.
   export NL_DX=${DX_CR}
   export NL_DY=${DX_CR}
#  export NL_START_DATE=${FILE_DATE}
#  export NL_END_DATE=${NEXT_FILE_DATE}
   ${HYBRID_SCRIPTS_DIR}/da_create_wps_namelist_RT.ksh
#
   if [[ -f job.ksh ]]; then rm -rf job.ksh; fi
   touch job.ksh
   RANDOM=$$
   export JOBRND=geogrid_${RANDOM}
   rm -rf *.jerr
   rm -rf *.jout
   cat << EOF >job.ksh
#!/bin/ksh -aeux
#BSUB -P ${PROJ_NUMBER_ACD}
#BSUB -n 1                                  # number of total (MPI) tasks
#BSUB -R "span[ptile=${GEOGRID_TASKS_PER_NODE}]"   # mpi tasks per node
#BSUB -J ${JOBRND}                          # job name
#BSUB -o ${JOBRND}.jout                     # output filename
#BSUB -e ${JOBRND}.jerr                     # error filename
#BSUB -W ${GEOGRID_TIME_LIMIT}              # wallclock time (minutes)
#BSUB -q geyser
#
mpirun.lsf ./geogrid.exe  > index.html 2>&1
#
export RC=\$?
if [[ -f SUCCESS ]]; then rm -rf SUCCESS; fi
if [[ -f FAILED ]]; then rm -rf FAILED; fi
if [[ \$RC = 0 ]]; then
   touch SUCCESS
else
   touch FAILED
   exit
fi
EOF
   bsub -K < job.ksh
fi
#
#############################################################################
#
# RUN UNGRIB
#
#############################################################################
```

Modular code block controlling the WPG GEOGRID task execution. This code block marks the beginning of the task execution part of the run script. Subsequently task are similarly identified. In general, it should not be necessary for the user to change these code blocks.

Arthur P. Mizzi
mizzi@ucar.edu

The PANDA_REAL_TIME_DATA Input Directory

mizzi on cheyenne6: /glade/p/FRAPPE/PANDA_REAL_TIME_DATA

```
total 1536
drwxr-sr-x  2 mizzi nacd0001    512 Apr  5  2017 airnow_csv_data/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 21 10:44 anthro_emissions/
drwxr-sr-x  4 mizzi nacd0001    512 Apr  5  2017 bio_emissions/
drwxr-sr-x  2 mizzi nacd0001    512 Jul  5  2017 china_epa_data/
drwxr-sr-x  6 mizzi nacd0001 131072 Apr  5  2017 fire_emissions/
drwxr-sr-x 14 mizzi nacd0001 131072 Dec  5 12:54 gfs_forecasts/
drwxr-sr-x  2 mizzi nacd0001    512 Apr  5  2017 hist_io_files/
drwxr-sr-x  3 mizzi nacd0001    512 Apr  5  2017 iasi_co_hdf_data/
drwxr-sr-x  2 mizzi nacd0001 131072 Apr  5  2017 iasi_o3_hdf_data/
drwxr-sr-x 18 mizzi nacd0001 131072 Apr  5  2017 met_obs_prep_data/
drwxr-sr-x  5 mizzi nacd0001    512 Apr  5  2017 modis_aod_hdf_data/
drwxr-sr-x  2 mizzi nacd0001    512 Apr  5  2017 mopitt_co_hdf_data/
drwxr-sr-x  2 mizzi nacd0001    512 Jul  4  2017 mozart_forecasts/
drwxr-sr-x  2 mizzi nacd0001    512 Apr  5  2017 static_files/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 19:49 wes_coldens/
cheyenne6:/glade/p/FRAPPE/PANDA_REAL_TIME_DATA>
```

Test case input directory listing. Generally, the subdirectories contain the raw input data in the formats used by WRF-Chem/DART. Some of the subdirectories contain the files and run scripts needed to create the WRF-Chem static input files.

Arthur P. Mizzi
mizzi@ucar.edu

The PANDA_REAL_TIME_DATA Output Directory

mizzi on cheyenne6: /glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST

```
total 512
drwxr-sr-x 19 mizzi nacd0001 131072 Nov 14 17:45 2014072418/
drwxr-sr-x 24 mizzi nacd0001 131072 Nov 14 17:44 2014072500/
drwxr-sr-x  2 mizzi nacd0001    512 Sep 15 15:01 geogrid/
-rw-r--r--  1 mizzi nacd0001    320 Sep 16 10:49 pert_file_emiss
-rw-r--r--  1 mizzi nacd0001     95 Sep 16 10:46 pert_file_icbc
cheyenne6:/glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST>
```

Test case output directory listing. Generally, the date/time subdirectories contain the cycle out (examples are given below), and the "pert_file_xxx" subdirectoriescontain data for generating the emission (_emiss) and initial/boundary condition (_icbc) ensemble members.

mizzi on cheyenne6: /glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST/2014072418

```
total 3328
drwxr-sr-x  2 mizzi nacd0001    512 Sep 17 10:28 combine_obs/
drwxr-sr-x  2 mizzi nacd0001    512 Sep 15 20:19 exo_coldens/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 15:26 metgrid/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 10:06 mopitt_co_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 10:28 prepbufr_met_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 10:28 preprocess_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 15:28 real/
drwxr-sr-x  2 mizzi nacd0001    512 Sep 15 20:19 seasons_wes/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 15:26 ungrib/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 21:33 wrfchem_bio/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 21 11:26 wrfchem_chem_emiss/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 16 08:12 wrfchem_chemi/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 16 10:49 wrfchem_chem_icbc/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 16 10:46 wrfchem_fire/
drwxr-sr-x 12 mizzi nacd0001    512 Sep 17 10:29 wrfchem_initial/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 15 15:55 wrfchem_met_bc/
drwxr-sr-x 22 mizzi nacd0001 131072 Sep 15 15:51 wrfchem_met_ic/
cheyenne6:/glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST/2014072418>
```

2014072418 – "Initial Cycle" subdirectory listing

mizzi on cheyenne6: /glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST/2014072500

```
total 4352
drwxr-sr-x  2 mizzi nacd0001    512 Sep 17 13:24 combine_obs/
drwxr-sr-x 22 mizzi nacd0001 131072 Sep 20 14:53 dart_filter/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 18 18:36 ensemble_mean_input/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 18 18:38 ensemble_mean_output/
drwxr-sr-x  2 mizzi nacd0001    512 Sep 17 12:15 exo_coldens/
drwxr-sr-x  2 mizzi nacd0001 131072 Nov 14 17:42 metgrid/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:19 mopitt_co_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:23 prepbufr_met_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:24 preprocess_obs/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 11:58 real/
drwxr-sr-x  2 mizzi nacd0001    512 Sep 17 12:15 seasons_wes/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 11:56 ungrib/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 18 15:08 update_bc/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:10 wrfchem_bio/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 21 11:22 wrfchem_chem_emiss/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 21 10:49 wrfchem_chemi/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:17 wrfchem_chem_icbc/
drwxr-sr-x 12 mizzi nacd0001    512 Sep 18 17:24 wrfchem_cycle_cr/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 13:13 wrfchem_fire/
drwxr-sr-x  2 mizzi nacd0001 131072 Sep 17 12:15 wrfchem_met_bc/
drwxr-sr-x 22 mizzi nacd0001 131072 Sep 17 12:11 wrfchem_met_ic/
cheyenne6:/glade/p/FRAPPE/real_PANDA_RETR_VARLOC_RELEASE_TEST/2014072500>
```

2014072418 – "First Assimilation Cycle" subdirectory listing

Arthur P. Mizzi
mizzi@ucar.edu

Ensemble Kalman Filter

- Mitchell and Houtekamer (2001):

$$\boldsymbol{x^a = x^b + K\left(y^o - Hx^b\right)}$$

where $\boldsymbol{K = PH^T(HPH^T + R)^{-1}}$ - $\boldsymbol{P}$ is the ensemble error covariance, $\boldsymbol{R}$ is the observation error covariance, and $\boldsymbol{H}$ is the forward operator mapping state variable $\boldsymbol{x}$ to observation $\boldsymbol{y}$.

- **Problem:** undersampling can lead to spurious correlations in $\boldsymbol{P}$.
- **Problem:** overfitting and cycling can lead to degeneracy/spread collapse.
- **Solution:** Localization – limit the spatial extent and/or state variable/observation correlations (reduce or set them to zero).
- **Solution:** Increase observation error to avoid overfitting
- **Solution:** Use prior or posterior covariance inflation.

Ensemble Kalman Filter Least Squares Framework - Spread Collapse

$$\Delta x = \left(\frac{cov(x,y)}{var(y)}\right)^{1/2} \times \Delta y$$

x - the ensemble of state variables at a grid locations

y - the ensemble of expected observations at an observation location

$y = H(x)$ where H is the forward operator

Δy - the ensemble of expected observation increments

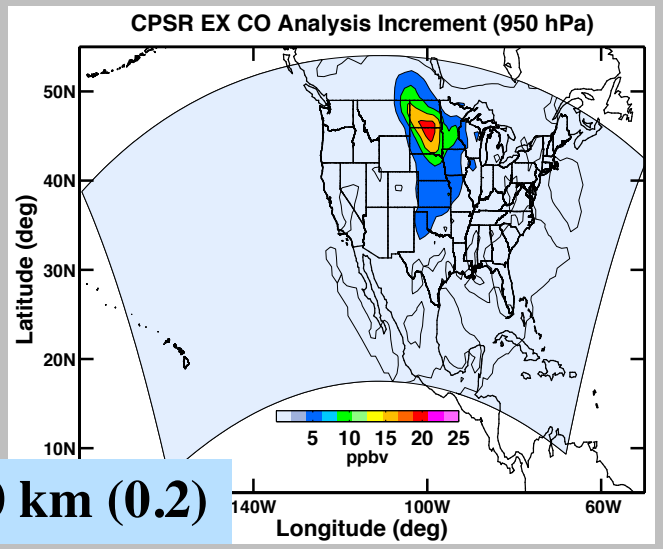
Δx - the ensemble of state variable increments

With spread collapse $var(y) \rightarrow 0$

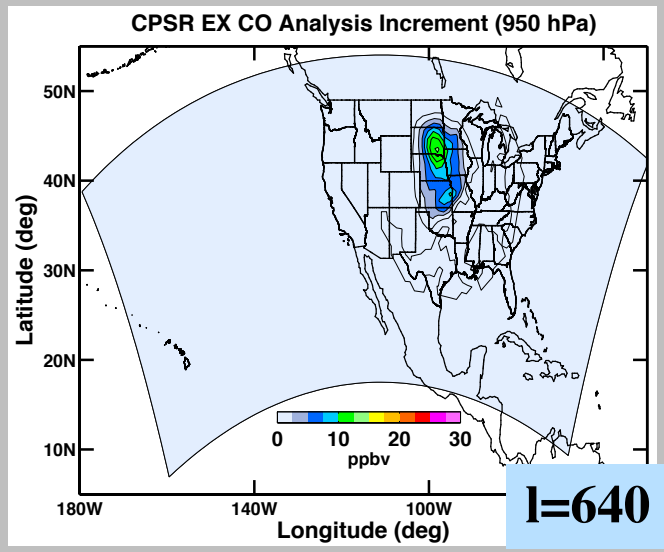
Anderson (2003)

Arthur P. Mizzi
mizzi@ucar.edu

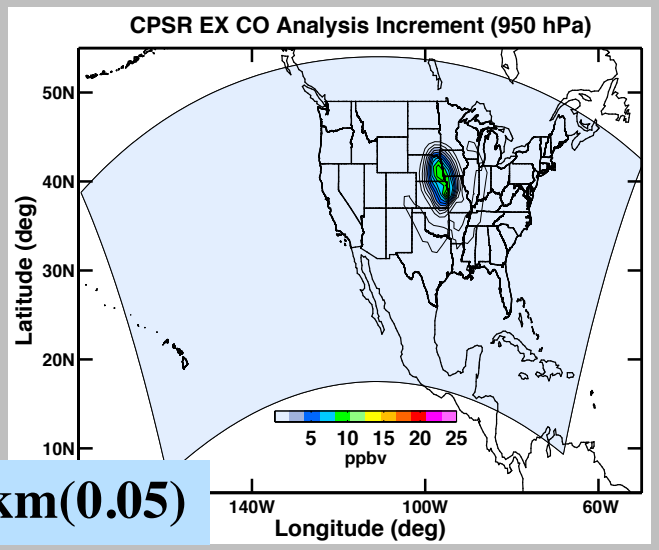
WRF-Chem/DART: Localization Examples



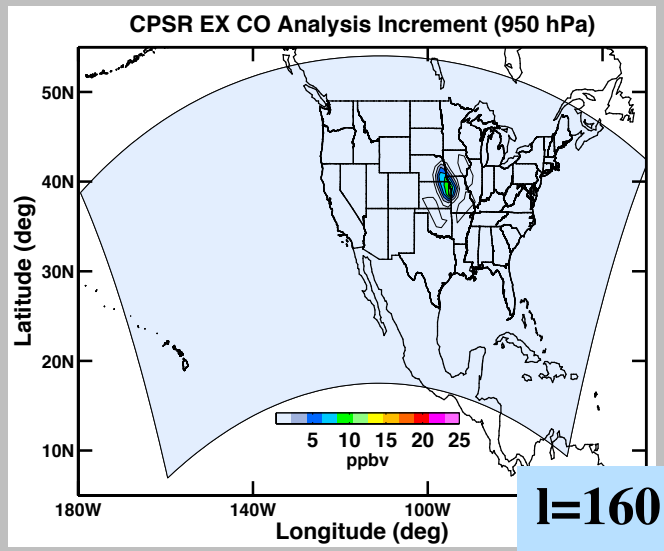
l=1300 km (0.2)



l=640 km (0.1)



l=320 km(0.05)



l=160 km (0.025)

Arthur P. Mizzi
mizzi@ucar.edu

References:

- Mizzi, A. P., A. F. Arellano, D. P. Edwards, J. L. Anderson, and G. G. Pfister (2016): Assimilating compact retrievals of atmospheric composition with WRF-Chem/DART: A regional chemical transport/ensemble Kalman filter data assimilation system. *Geosci. Model Dev*., 9, 1-14.
- Mizzi, A. P., D. P. Edwards, and J. L. Anderson (2018): Assimilating compact phase space retrievals (CPSRs): Comparison with independent observations (MOZAIC *in situ* and IASI retrievals) and extension to assimilation of retrieval partial profiles. [*under internal review*].

Arthur P. Mizzi
mizzi@ucar.edu

Additional References:

- Lahoz, W., B. Khattatov, and R. Menard, 2010: Data Assimilation, Making Sense of Observations. Springer-Verlag, ISBN 978-3-540-74702-4.
- Kalnay, E., 2003: Atmospheric Modeling, Data Assimilation, and Predictability. Cambridge University Press, ISBN 0-521-7917940.
- Daley, R., 1991: Atmospheric Data Assimilation. Cambridge University Press, ISBN 0-521-38215-7,
- Kuo, Y.-H. and T. W. Schlatter, 1990: Mesoscale Data Assimilation. 1990 NCAR/ASP Summer Colloquium.

Arthur P. Mizzi
mizzi@ucar.edu